Effect of Various Activation Function on Steering Angle Prediction in CNN based Autonomous Vehicle System

Rounak Hazra, Aman Kumar, B. Baranidharan

Abstract: Autonomous or Self-driving vehicles are set to become the main mode of transportation for future generations. They are highly reliable, very safe and always improving as they never stop learning. There are numerous systems being developed currently based on various techniques like behavioural cloning and reinforcement learning. Almost all these systems work in a similar way, that is, the agent (vehicle) is completely aware of its immediate surroundings and takes future decisions based on its own historical experiences. The proposed work involves the design and implementation of Convolutional Neural Network (CNN) enhanced with new activation function. The proposed CNN is trained to take a picture of the road in front of it as input and give the required angle of tilt of the steering wheel . The model is trained using the behavioural cloning method and thus learns to navigate from the experiences of a human agent. This method is very accurate and efficient. In this paper, for the detection of object and vehicle in autonomous vehicle, the existing Tensorflow object Detection API is collaborated with pretrained SSD MobileNet model. This paper presents in detail literature survey on various techniques that have been used in predicting steering angle and object detection in self driving car. Apart from that, the effect of activation functions like ReLU, Sigmoid and ELU over the CNN model is analysed.

Keywords: Autonomous driving vehicle, Residual Net, Convolutional Neural Network , Activation function .

I. INTRODUCTION

An autonomous vehicle is a vehicle that is capable of sensing its environment and navigating without human control [1]. (en. Wikipedia. Org /wiki /Autonomouch) There are many different approaches being employed to successfully navigate to the destination. Behavioural cloning, Reinforcement Learning and more specifically Q-Learning are the most common techniques being used to implement these autonomous vehicle systems. Behavioural cloning, as the name suggests, clone the actions of some other entity to learn the process. In the case of autonomous vehicles, the behavioural cloning agent learn the driving cues from observing a human driving actions. Various parameters are extracted from the human including the velocity, throttle, like and steering wheel angle in addition to the images of the road captured through externally mounted cameras.

Other sensors such as LIDAR, RADAR, proximity sensor and infrared camera are also be used in addition to these sensors. CNN is used to execute the vectorized images that are fed into the autonomous vehicle system. The expected output from the CNN is the steering wheel angle. After sufficient training and subsequent counting, the CNN is ready to take only a single vectorized image as input and predict the proper angle of the steering wheel in order for the autonomous vehicle to remain on course. Object detection in autonomous vehicle is by using object detection algorithm interface in Tensorflow, which is used along with modelling of pretrained SSD MobileNet. Mobile-Nets is efficient for mobile devices with TensorFlow model of mobile devices. The proposed system uses the angle value of steering and detecting object and vehicle motion, where the distance among the cars is relative and is used under vehicle commands by making the correct decision. Then, the analysis is done on the steering wheel movement based on the angle of steering movement. Fig. 1 depicts the general steering angle prediction model in autonomous driving vehicles.

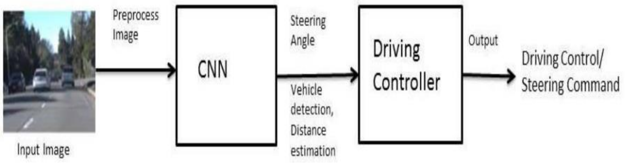

Preprocess Image
Input Image
CNN
Steering Angle
Vehicle detection, Distance estimation
Driving Controller
Output
Driving Control/ Steering Command

Fig. 1: Steering angle prediction model

II. RELATED WORK

Sumit Joshi, Narayan Pawar [1] created End to End architecture for autonomous vehicle detection and object algorithm for driving paradigm. Their work mainly concentrated on estimating driving actions based on blind mapping or direct image commands.

Rodolfo Valiente , Mahdi Zaman [2] proposed Deep learning based End to End CNN network for complete and total utilization of vehicle assist and steering problems. Autonomous sharing between drivers and Udacity data set is used for their experimentation.

Shuyang Du , Hauli Gao [3] have developed their model using #D modelling of convolution layer for LSTM recurrent layer, and data set learning by transfer learning.

Aatiq Oussama and Talea Mohamed [4] have done concluded that the combination of ResNet 50 architecture with event cameras can lead to better steering angle prediction.

Revised Manuscript Received on December 30, 2019.
* Correspondence Author

Rounak Hazra*, Pursuing, B.tech , C.S.E , SRM Institute Of Technology and Science, Chennai . Email: rounakhazra_ranjit@srmuniv.edu.in

Aman kumar , Pursuing, B.tech ,C.S.E , SRM Institute Of Technology and Science , Chennai . Email: amankumar_anil@srmuniv.edu.in

Dr. B. Baranidharan, Associate Professor, S.R.M Institute of science and technology, Chennai.

The authors uses large number of computer vision based approaches like Euclidean method, CNN LSTM model and ResNet 50 architecture model. The ResNet 50 architecture model gives the best prediction for steering angle .

NehaYadav and RishiMody [5] has designed CNN based on VGG16 model and Hybrid model for solving the steering angle issue.

Zhilu Chen and Xinming Huang [6] proposed an learning approach under end to end mechanism to predict image data for steering angle to maintain the self driving car in lane. They used comma.ai data set to train and evaluate the model which has the framework of steering angle details and data sets.

Djork Arne Clevert and Thomas Unterthiner [7] has utilized the Exponential Linear Unit (Clevert et al. 2015) activation function for the Convolution layer. It is used as an alternative to the ReLU function. Generally, ReLU function does not average to 0 and hence introduces a bias to consecutive convolutional layers. Also an average for the activation function ensures faster learning rate.

Anish Shah and Sameer Shinde [8] followed linear unit exponential approach in Residual networks. In their work, they achieved increased efficiency and speed in depth accuracy of residual learning using CIFAR – 10 and CIFAR – 100 categories.

Mariusz Bojarski , Davide DelTesta , Daniel Dworakowski , Bernhard Firner[9] follows a powerful approach using a convolutional neural network for mapping raw pixels. Automatic traffic detection is performed by end to end human features for the necessary processing steps . The human steering angle act as the training signal to detect useful road feature. It used an NVIDIA DevBox,Torch 7,NVIDIA DRIVEPX for training and to determine at which direction to drive the vehicle. System operated at 30 frames per second. Certain work is needed to refine robustness of network, verify it and improve visualisation of processing steps.

Truong-Dong Do , Minh-Thien Duong [10] Proposed a Rasberry Pi monocular vision based self-driving car prototype using CNN. The vehicle’s top speed is about 5-6 km/hr in various driving condition. Twofold contributions for mapping and prediction of CNN. Usage of perfect synchronisation in time environment, and, tracks. Including real time and embedded systems under neural networks. The major drawback is the latency of the camera.

Xin Zhang, Maolin Chen [11], developed cognitive and adaptive model for neural networks for development and decision making for driverless navigation. Main features include cloning of behaviour transfer learning and simulation.

Yue Kang , Hang Yi [12] proposed different type of datasets that are used for self driving car model . Looping approach algorithm is used for self driving. Included additional datasets and virtual testing survey using simulation approach. 37 open testing datasets and 20 virtual testing environment are surveyed in detail.

Michael G. Bechtel†, Elise McEllhiney [13]: It is a low cost autonomous car platform based on deep neural network . Self state of the art evaluation to protect for protection of class partitioning and memory throttling. DeepPicar’s network architecture—9 layers, 27 million connections and 250K parameters—and can drive itself in real-time using a web camera and a Raspberry Pi 3 quad-core platform.

Vijay John , Ali Boyali [14] developed inclusive algorithm for autonomous driving using a monocular camera individual distributions and filtering patterns. Modelling recurrent convolutional Network and accurate particle filters for autonomous and particle driving including pattern analysis and neural network.

Henryk Blasinski1, Joyce Farrell [15] Uses of open source free software and simulating environment for Ray tracing and quantitative computation. Transformation of irradiance accounting the responses by censor pixel and its ability to preprocess the images.

From the above literature, we observed that none of the research focussed on applying and evaluating different activation functions. In the proposed system a new CNN model is built and tested with ReLU, Sigmoid and ELU activation functions.

III. PROPOSED METHODOLOGY

In the proposed work, an End-to-End CNN based model is verified using different activation function like Elu, Relu and Sigmoid. The main focus of our research is to find the best activation function for deep learning model for the self-driving car simulator . Also, we want to find the best model for object location and detection in less cost. MobileNet V1 is one of the best model for object location and detection at cheaper cost.

CNN (CONVOLUTIONAL NEURAL NETWORK)

The Convolutional Neural Network (CNN) is the standard deep learning algorithm used for computer vision problems. Since 2012, their popularity is rising immensely due to AlexNet, which showcased its usefulness in solving computer vision problems using only 8 layers. CNNs are very easy to deploy as they do most of the work intuitively and require very little parameterized input from the user to tune the neural network. It can intuitively select features based on its importance and does not need the user to select the features. The Convolutional Neural Network is both temporally and computationally much more efficient than other traditional deep learning algorithms. Thus these systems are deployed on a wide range of platforms, even certain low cost ones like embedded systems and mobile devices, since their computational requirements are supported by most platforms. CNNs have a standard design and the input is usually an image. Convolutional layer, Max Pooling layer and Connected layers are the three types of layers used in CNNs. Convolutional layer precedes over the Max pooling and Connected layers follow the Max Pooling layer. Convolutional layers as the name suggests form the backbone of the Convolutional Neural Network. The Convolution operations are the mathematical way of combining 2 numerical values.

In the CNNs, as shown in Fig. 2, the input image is convoluted with the kernel using the convolutional mathematical operation.

The result of this is called a feature map. The kernel is called a convolution filter.

The following steps are incorporated in steering angle detection mechanism:

(1) Image collection and balancing data: Data collected from Udacity dataset and data is balanced using shuffle pre-processing technique.

(2) Data pre-processing using data augmentation methods: Using Keras, batch generators and fit generators data is read from the csv file and generating duplicate images using data augmentation technique .

(3) Creating CNN model for steering angle prediction: The final stage in steering angle prediction is creating CNN model using different activation function to identify which is the best.

(4) Testing various activation function: The experiment is to test different activation function to find the best model for self-driving car.

IV. EXPERIMENTS,RESULT AND DISCUSSIONS

Datasets

The data is collected from a free to use open source simulator as published by Udacity (https://github.com/udacity/self-driving-car-sim). The simulator records the events from the perspective of 3 virtual cameras mounted on the front of the bonnet and the 2 sides of the car. It takes picture and the stores them in a csv file along with 4 other parameters. The four parameters are the steering wheel angle, throttle or acceleration, velocity and the brake applied. It samples all these values throughout the duration of the recording. Training data contains 4053 images . The steering angle corresponds to radian value range from -1 to +1 . When steering angle is straight value is 0, left is -1 and right is +1.

Data augmentations

Overfitting of data is avoided by data augmentation techniques. In our model, there are 4 levels of data augmentation. In first level, zoom augmentation is done in which parameters from 1 to 1.3 is set to improve feature extraction. In second level, horizontal and vertical shift augmentation is done with x at range from (-0.1 to 0.1) and y at range from (-0.1 to 0.1).In third level augmentation, brightness augmentation is done. To intensify the pixel and making darker images. The range between 0.2 and 1.2 is set. In fourth level, horizontal flip augmentation is done.

Preprocessing

The input image shape is 160x32. Using data augmentation techniques and Keras's lambda layer cropped vertically the image to 88x320x3. Image is intensified by -0.5 to 0.5. Keras image resize function resized the image to 66x200x3.

Models

It processed the csv file that contains the image file names and their corresponding steering angle and then randomized the data and split into 80/20 ratio. 20% of the data is used for validation purpose. All our models are trained and evaluated in GPU. For complete end to end learning, the CNNs are modelled after the actual neural network used for practical purposes by the Defence Advanced Research Project Agency. The proposed model has a total of 9 layers and is built using pre-defined Keras functions. It use the exact same model that is used in real world applications. The layers are, as specified by researchers in Nvidia in Fig. 3. This model consists of 5 convolutional layers with Max pooling between them followed by a couple of dense layers and a linear activation to output continuous steering angles. The dataset is split into 80:20 form and subsequent testing is done. Mean Squared Error (MSE) loss function is used for the model and it is trained for 10 epochs. Fig. 2 depicts the proposed CNN architecture.

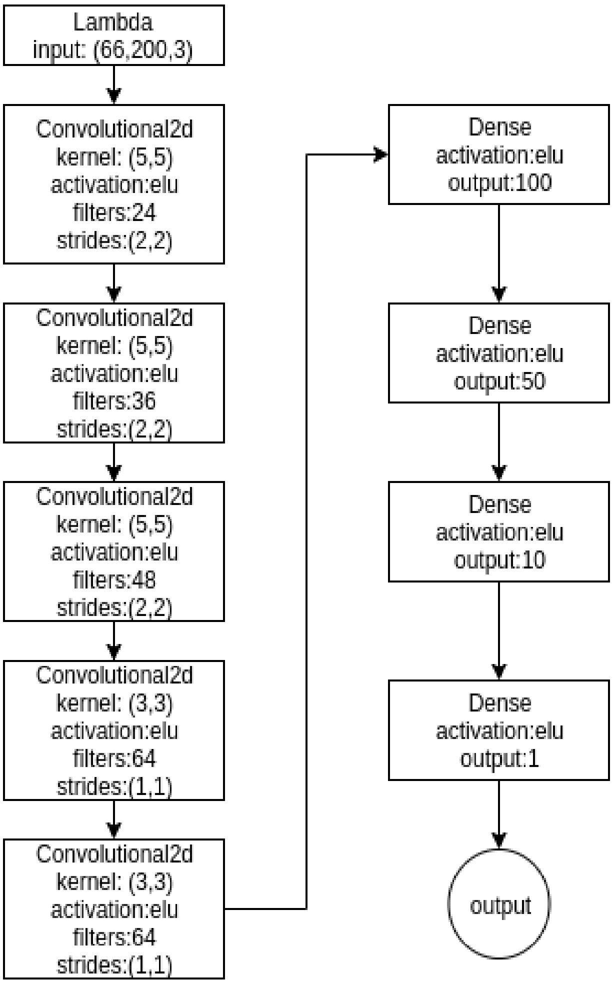


Fig. 2: Proposed CNN model

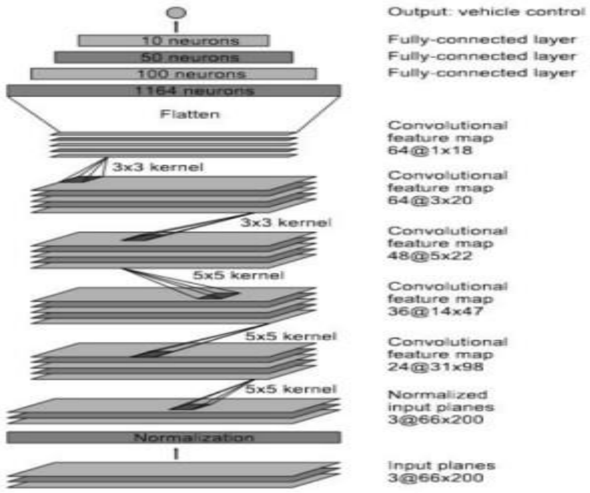


Fig. 3: Nvidia architecture [9]

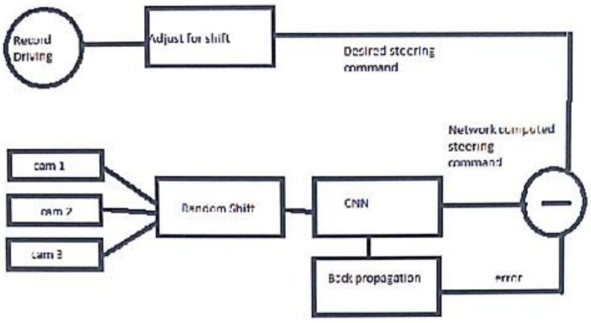


Fig. 4: Proposed System architecture

Testing with different activation functions

ReLU: The relu activation function is tested with the model and different weight initialization combination methods. The ReLU function does not average to 0 and hence introduces a bias to consecutive convolutional layers. Also an average for 0 for activation function insures faster learning rate. ReLU often gets stuck at a point from where its value is constant. For training set, the loss value is 0.1171 and for the validation set, the value is 0.1160. The graph is not linear. Relu has dying problem. Fig. 5 captures the training and validation loss for ReLU function.

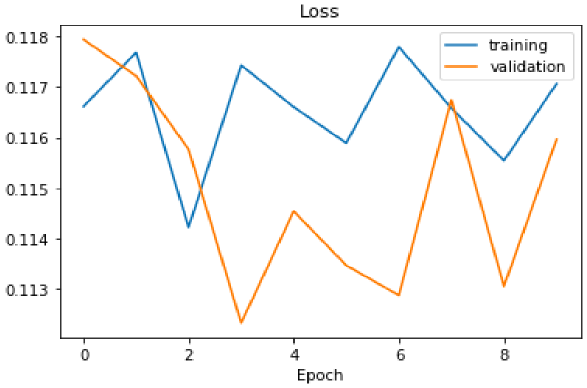


Fig. 5: Training and Validation loss for ReLU

Sigmoid: Undesirable property of sigmoid function is that, it gets saturates at the end of 0 or 1. The graph is non-linear and sigmoid outputs are zero centred. Gradient updates of weights are present with undesirable zig zag dynamics. If the weights are too large, then most neurons would become saturated and network will barely learn. Th training set loss value is 0.1158 and validation set loss is 0.1095. Fig. 6 depicts the training and validation loss for sigmoid function. But the validation loss improved compared with Relu activation function.

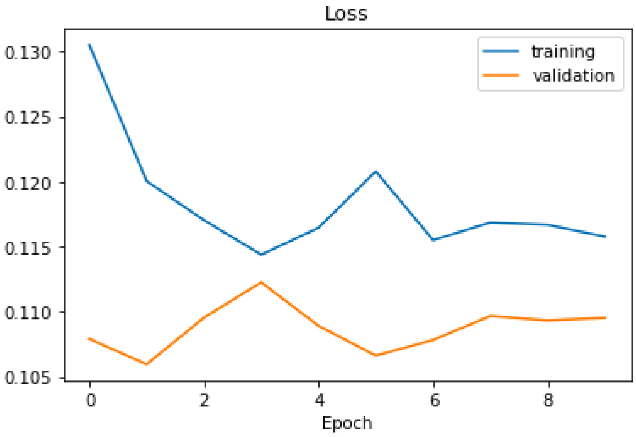


Fig. 6: Training and Validation loss for Sigmoid

Elu: Exponential Linear Unit (Clevert et al. 2015) is the activation function chosen for use in the Convolutional layers. It is used as an alternative to the ReLU function. Then ReLU function does not average to 0 and hence introduces a bias to consecutive convolutional layers. Also an average for the activation function ensures faster learning rate. The Exponential Linear Unit rectified this problem. Its average is close to 0. It also takes care of the vanishing gradient problem. ReLU often gets stuck at a point from there its value is constant. Fig. 7 shows the training and validation loss for ELU function. Exponential Linear Unit does not face this issue as it is an exponential function .

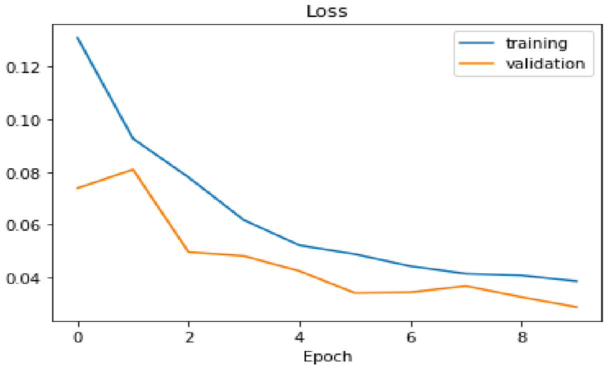
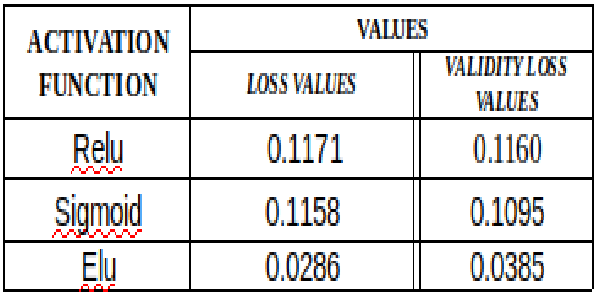


Fig. 7 Training and validation loss for Elu

V. RESULTS

Table- I Loss value and validity loss values of different activation function



ACTIVATION FUNCTION	VALUES	
	LOSS VALUES	VALIDITY LOSS VALUES
Relu	0.1171	0.1160
Sigmoid	0.1158	0.1095
Elu	0.0286	0.0385

From the experimental results it is observed that the ELU activation function is best for CNN model in combination with Adam optimizer. ReLU is constant for the negative values but ELU gradually attains its final value of negative alpha. It is easy to find the derivative of the ELU function. The derivative is required for backpropagation and hence having a function with low computational and memory requirements make the calculations fast to compute.

VI. CONCLUSION

Steering angle prediction is an important topic in building the Autonomous Driving Vehicles. With the improvement in computation power, CNNs are becoming popular for computer vision problems. Apart from number of convolutional layers, Max Pooling layers and optimizer, activation function plays a major role in reducing the loss of the CNNs. Thus in this paper, the proposed CNN is experimented with three activation functions such as ReLU, Sigmoid and ELU. Generally, ReLU does not average to 0 and introduces errors which are propagating to the subsequent layers and affect the overall performance. In sigmoid, saturation is the major problem which diminishes the learning capability of neurons. But, it is found out from the experiments that ELU reduces both the training and validation loss compared with ReLU and Sigmoid. Also, ELU increases the learning capability of the neurons. In future, more such deep learning models with optimized number of layers will be tested with ELU function for analysing its effectiveness.

REFERENCE

1. Sumit Joshi , Narayan Pawar , Vivek Sonara , Sudhir Sul , Prof. S. A. Mulay " End-To-End aDriving Controls Prediction From Images Using CNN " International Advanced Research Journal in Science, Engineering and Technology ISO 3297:2007 Certified Vol. 5, Issue 3, March 2018 .
2. Rodolfo Valiente , Mahdi Zaman , Sedat Ozer , Yaser P. Fallah "Controlling Steering Angle for Cooperative Self-driving Vehicles utilizing CNN and LSTM-based Deep Networks " Center for Research in Electric Autonomous Transport (CREAT), Orlando, FL †University of Central Florida, Orlando, FL {rvalienter90, mahdizaman}@knights.ucf.edu, sedatist@gmail.com, yaser.fallah@ucf.edu .
3. Shuyang Du , Hauli Gao , Andrew Simpson "Self Driving Car Steering Angle Prediction Based on Image Recognition " http://cs231n.stanford.edu/reports/2017/pdfs/626.pdf
4. Aatiq Oussama and Talea Mohamed "A literature review of steering angle prediction algorithms for Self-driving cars " nformation Processing Laboratory, Ben M'SikFaculty of Sciences, Hassan 2 Casablanca University, Morocco aatiqoussama@gmail.com, taleamohamed@yahoo.fr .
5. NehaYadav and RishiMody "PredictSteeringAnglesinSelf-DrivingCars" https://rmmody.github.io/pdf/682Project.pdf .
6. Zhilu Chen and Xinming Huang "End-to-End Learning for Lane Keeping of Self-Driving Cars " 2017 IEEE Intelligent Vehicles Symposium (IV) June 11-14, 2017, Redondo Beach, CA, USA .
7. Djork Arne Clevert , Thomas Unterthiner & Sepp Hochreiter "FAST AND ACCURATE DEEP NETWORK LEARNING BY EXPONENTIAL LINEAR UNITS (ELUS) " https://arxiv.org/pdf/1511.07289.pdf .
8. Anish Shah , Sameer Shinde , Eashan Kadam , Hena Shah , Sandip Shingade "Deep Residual Networks with Exponential Linear Unit " https://www.arxiv-vanity.com/papers/1604.04112/ .
9. Mariusz Bojarski , Davide DelTesta , Daniel Dworakowski , Bernhard Firner , Beat Flepp , Prasoon Goyal , LawrenceD . Jackel , Mathew Monfort , Urs Muller , Jiakai Zhang , Xin Zhang , Jake Zhao , Karol Zieba " EndtoEndLearningforSelf-DrivingCars " https://openreview.net/forum?id=SyyHSsL9 .
10. Truong-Dong Do , Minh-Thien Duong, Quoc-Vu Dang and My-Ha Le* " Real-Time Self-Driving Car Navigation Using Deep Neural Network " *2018 4th International Conference on Green Technology and Sustainable Development (GTSD)*, Ho Chi Minh City, 2018, pp. 7-12.
doi: 10.1109/GTSD.2018.8595590 .
11. Xin Zhang, Maolin Chen, Xingqun Zhan "Behavioral cloning for driverless cars using transfer learning" *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Monterey, CA, 2018, pp. 1069-1073. doi: 10.1109/PLANS.2018.8373488 .
12. Yue Kang , Hang Yin , and Christian Berger " Test Your Self-Driving Algorithm: An Overview of Publicly Available Driving Datasets and Virtual Testing Environments " IEEE TRANSACTIONS ON INTELLIGENT VEHICLES, VOL. 4, NO. 2, JUNE 2019 .
13. Michael G. Bechtel†, Elise McEllhiney†, Minje Kim, Heechul Yun " DeepPicar: A Low-cost Deep Neural Network-based Autonomous Car " https://arxiv.org/pdf/1712.08644.pdf.
14. Vijay John , Ali Boyali , Hossein Tehrani " Estimation of Steering Angle and Collision Avoidance for Automated Driving Using Deep Mixture of Experts " IEEE TRANSACTIONS ON INTELLIGENT VEHICLES VOL 3 NO. 4 DECEMBER 2018 .
15. Henryk Blasinski1, Joyce Farrell1, Trisha Lian1, Zhenyi Liu1,2, Brian Wandell1,3 " Optimizing Image Acquisition Systems for Autonomous Driving " https://pdfs.semanticscholar.org/2809/4e414200918e3b3fd0f7e5e3e9d29b092be4.pdf .

AUTHORS PROFILE

Rounak Hazra, C.S.E , B.tech ,
SRM Institute Of Technology and Science, Chennai .
Email: rounakhazra_ranjit@srmuniv.edu.in
Mob.no: 8778415818

Aman kumar, C.S.E , B.tech ,
SRM Institute Of Technology and Science ,
Chennai .
Email: amankumar_anil@srmuniv.edu.in
Mob.no: 8789880474

Dr.B.Baranidharan, has completed his Master of Technology in Computer Science and Engineering from SRM IST, Chennai and PhD in Wireless Sensor Networks (specialization) from SASTRA Deemed University, Thanjavur. Currently, he is working as Associate Professor in the department of CSE, SRM IST. He is having more than 10 years of academic experience and have published 22 papers in various International Journals and Conferences. Earlier his research involved about designing new clustering architecture for Wireless Sensor Networks and Internet of Things using various computational techniques. His current research includes Artificial Intelligence, Machine learning, Deep learning and Internet of Things.